Burst Statistics of Viterbi Decoding

L. J. Deutsch and R. L. Miller
Communications Systems Research Section

A mathematical model of Viterbi decoder burst error performance is presented. This model allows for computer generation of Viterbi-like error sequences quickly and inexpensively for applications where large amounts of data are required. The model was corroborated through comparisons with actual software decoder simulations.

I. Introduction

It is well known that the bit errors produced by Viterbi decoding are not at all independent. Instead, they tend to group together in error clumps known as "bursts." This happens because error events in a Viterbi decoder are caused by excursions from the correct path in the code trellis structure (Ref. 1). Hence the implementation of convolutional encoding with Viterbi decoding transforms a Gaussian channel (such as the deep space channel) into a fading channel (see Fig. 1).

The DSN has adopted convolutional coding as a standard for deep space missions. The standard code is a (7, 1/2) convolutional code, which is currently decoded by Viterbi decoders.

Until now, the preferred method for studying the Viterbi channel has been with actual Viterbi decoding hardware or software. Running Viterbi decoder hardware for the purposes of such studies can be much more expensive than computer simulation. On the other hand, the software approach is usually so slow as to be prohibitively expensive. In this article, a method is presented for producing Viterbi-like error sequences both quickly and inexpensively using Monte-Carlo techniques.

Software Viterbi decoder simulations (of the type described in Ref. 2) have shown that burst lengths, as well as the times between consecutive bursts (known as "waiting times"), are very nearly geometrically distributed. The parameters needed to define these distributions are the average burst length, $\overline{B}$, the average waiting time, $\overline{W}$, and the average density of errors in a burst, θ. Given these parameters, Viterbi decoder burst lengths, B, were observed to be distributed according to

$$pr(B = m) = p\,(1 - p)^{m-1} \qquad (m > 0) \tag{1}$$

where

$$p = 1/\overline{B}.$$

Errors within bursts occur randomly with probability θ. Waiting times, W, were observed to be distributed according to

$$pr\,(W = n) = q\,(1 - q)^{n-K+1} \qquad (n \geqslant K - 1) \tag{2}$$

where K is the constraint length of the code and

$$q = 1/(\overline{W} - K + 2)$$

This description of Viterbi decoder burst statistics is called the "geometric model."

II. Summary of Results

A Monte Carlo software routine was written to generate Viterbi error sequences directly from Eqs. (1) and (2). The advantage of doing this is that Viterbi decoder simulation software requires several orders of magnitude more calculations per decoded bit than such random number generation techniques. On the computer used for this study (an XDS Sigma-5), the software Viterbi decoder required about 2^{K-7} hours per million bits for a code of constraint length K, while the geometric model required an average of five minutes per billion bits.

In order to validate the geometric model of Viterbi burst error statistics, the Viterbi channel of Fig. 1 was embedded in a Reed-Solomon coding scheme as shown in Fig. 2. The Reed-Solomon code used is a (255, 223) code capable of correcting up to 16 8-bit symbol errors per codeword. This concatenated coding scheme is a proposed NASA standard for deep space missions (Ref. 3). Normally, the Reed-Solomon symbols would be interleaved to a depth of four or five so as to minimize the effects of Viterbi burst errors. In this article the symbols were not interleaved, in order to maximize the effects of the bursts. The resulting Reed-Solomon word and bit error probabilities were calculated by tabulating the errors generated using both a Viterbi software decoder and a Monte-Carlo routine that generated random bursts and waiting times according to the geometric model.

The results of these comparison runs are shown for various convolutional codes in Figs. 3 to 8. (Some of the curves exhibited in these figures run off the edge of the page since the next data point was too low to be plotted on the same scale.) At high signal-to-noise ratios (SNRs), fewer error events were observed, and hence the uncertainty in the results is higher at these points. Error bars indicating a 90% confidence interval are included in Fig. 5. It can be seen that the geometric model and the actual data agree to within the uncertainty of the experiments.

III. The Definitions of "Burst" and "Waiting Time"

Denote the constraint length of the convolutional code under consideration by K. Consider a sequence of bits output by the Viterbi decoder of the form

$$\overbrace{ccc\ldots c}^{K-1}\;\overbrace{e\,xxx\ldots x\,e}^{B}\;\overbrace{ccc\ldots c}^{K-1}$$

where the letter c represents a correctly decoded bit, an e represents a bit error, and an x may be either correct or in error. Suppose also that there is no string of $K - 1$ consecutive c's in the sequence $xxx \ldots x$. Then the string $exxx \ldots xe$ is called a "burst" of length B. The motivation behind this definition of a burst is that a string of $K - 1$ consecutive correct bits will return the Viterbi decoder to the correct decoding path. A string of c's between two bursts will be referred to as a "waiting time."

IV. Derivation of the Geometric Model of Burst Statistics

A random variable X is said to be geometrically distributed with parameter $p \epsilon [0, 1]$ if

$$pr\,(X = s) = p\,(1 - p)^s \qquad (s = 0, 1, 2, \ldots)\,.$$

For the purposes of this section, a random variable Y satisfies a "modified geometric distribution" of parameter $p \epsilon [0, 1]$ if there exists a positive integer d such that

$$pr\,(Y = s) = p\,(1 - p)^{s-d} \qquad (s = d, d + 1, d + 2, \ldots)\,.$$

In this case, Y will be called d-geometrically distributed.

It is shown in Ref. 4 by a random coding argument that burst lengths for an "average convolutional code" have a distribution that may be upper-bounded by a 1-geometric distribution. In this report, it will be shown that for convolutional codes of constraint lengths seven through ten, burst lengths are, in fact, very nearly 1-geometrically distributed. Moreover, the waiting times are $(K - 1)$-geometrically distributed.

The tests that were used to exhibit these facts were essentially the same for burst lengths and waiting times. For this reason, only the test for burst lengths will be described below.

Suppose that a software Viterbi decoder simulation is performed and N bursts are observed. Let B_i be the length of the i^{th} burst ($i = 1,2,3, \ldots, N$). Let B be the random variable representing burst length (so B_i is the i^{th} sample of the random variable B). It must be shown that

$$pr(B_i = s) = p(1 - p)^{s-1} \qquad (s = 1,2,3, \ldots)$$

for some $p \epsilon [0,1]$. The fact that these probabilities must sum to one forces $p = 1/\overline{B}$.

For each $m = 1,2,3, \ldots$ let N_m be the number of bursts of length greater than or equal to m. If the burst lengths were indeed 1-geometrically distributed with parameter $1/\overline{B}$, then the expected value of N_m/N_n would be

$$E(N_m/N_n) = \left(\sum_{s=m}^{\infty} p(1-p)^{s-1}\right) \Bigg/ \left(\sum_{s=n}^{\infty} p(1-p)^{s-1}\right)$$

$$= (1-p)^{m-n} = \left(1 - \frac{1}{\overline{B}}\right)^{m-n}.$$

In other words, for N sufficiently large,

$$1 - 1/\overline{B} \approx (N_m/N_n)^{1/(m-n)}$$

Since N is only moderately large in the software simulations that were performed for this study (on the order of 200 to 500), the performance of this test can be improved by grouping bursts of several consecutive lengths into bins. Enough bursts were placed into each bin so that $1 - 1/\overline{B}$ could be approximated to within 0.05 with 90% accuracy for each bin. These approximations remained reasonably constant between bins, indicating a successful test.

As remarked in Section II, waiting times were found, by a similar test, to be $(K - 1)$-geometrically distributed with parameter $q = 1/(\overline{W} - K + 2)$, where $\overline{W}$ is the average waiting time and K is the constraint length of the code.

The geometric model of Viterbi burst error statistics states that these bursts occur randomly according to these two modified geometric distributions. Errors within a burst occur essentially randomly (except for the fact that each burst starts and ends with an error) with probability θ. To generate error sequences similar to those produced by a Viterbi decoder, only the quantities $\overline{B}$, $\overline{W}$, and θ must be known. These are tabulated for several codes and channel SNRs in Tables 1 to 4.

References

1. McEliece, R. J. *The Theory of Information and Coding*, Addison-Wesley, London, 1977.

2. Butman, S. A., Deutsch, L. J., and Miller, R. M., "Performance of Concatenated Codes for Deep Space Missions," *TDA Progress Report 42-63, March and April 1981,* Jet Propulsion Laboratory, Pasadena, Calif., June 15, 1981.

3. Hooke, A. J., "Multimission Packet Telemetry Guidelines and Standards for Deep Space Missions," Document No. 663-9, Jet Propulsion Laboratory, Pasadena, Calif., November 1979 (an internal document).

4. Viterbi A. J., and Omura, J. K., *Principles of Digital Communication and Coding*, McGraw-Hill Book Company, New York, 1979.

Table 1. Viterbi decoder burst statistics, 3233013 (7, 1/2) convolutional code

E_b/N_0, dB	$\overline{B}$	$\overline{W}$	θ
0.5	25.84	131.2	0.564
0.75	23.46	158.5	0.566
1.0	21.07	220.5	0.571
1.1	19.78	275.8	0.574
1.2	19.27	293.2	0.574
1.3	18.02	371.2	0.573
1.4	17.46	430.6	0.573
1.5	17.01	474.1	0.578
1.6	15.76	600.8	0.578
1.7	15.21	702.2	0.579
1.8	14.32	847.0	0.586
1.9	13.50	931.7	0.584
2.0	12.89	1122	0.590
2.5	10.17	3258	0.599
3.0	8.67	9596	0.584
3.5	6.70	3.7E4	0.630
4.0	4.40	2.0E5	0.591

Table 2. Viterbi decoder burst statistics, 7376147 (7, 1/3) convolutional code

E_b/N_0, dB	$\overline{B}$	$\overline{W}$	θ
0.5	16.80	228.3	0.596
0.6	15.79	258.6	0.598
0.7	15.31	290.1	0.601
0.8	14.70	308.2	0.602
0.9	13.94	355.5	0.605
1.0	13.24	440.1	0.612
1.1	13.13	473.5	0.611
1.2	12.13	567.1	0.613
1.3	12.01	663.4	0.615
1.4	11.40	787.2	0.620
1.5	11.30	980.8	0.624
1.6	10.79	1146	0.622
2.0	9.46	2556	0.636
2.5	7.53	8613	0.653
3.0	6.35	2.9E4	0.685
3.5	7.25	1.2E5	0.672

Table 3. Viterbi decoder burst statistics, 3103320323 (10, 1/2) convolutional code

E_b/N_0, dB	$\overline{B}$	$\overline{W}$	θ
0.5	37.98	162.4	0.511
0.6	35.99	184.8	0.512
0.7	32.72	221.7	0.517
0.8	30.11	248.5	0.515
0.9	28.07	292.9	0.518
1.0	26.98	353.0	0.523
1.2	25.16	526.7	0.530
1.3	22.86	601.0	0.530
1.4	21.15	857.6	0.537
1.5	21.13	983.6	0.531
1.6	20.86	1217	0.545
1.7	18.80	1566	0.541
2.0	16.95	4048	0.551
2.5	14.14	2.5E4	0.585
3.0	11.25	2.5E5	0.622

Table 4. Viterbi decoder burst statistics, 7461776427 (10, 1/3) convolutional code

E_b/N_0, dB	$\overline{B}$	$\overline{W}$	θ
0.5	25.29	398.1	0.533
0.6	24.84	455.3	0.532
0.7	22.06	549.4	0.539
0.8	21.37	642.4	0.541
0.9	20.76	813.0	0.540
1.0	19.34	990.1	0.540
1.2	17.68	1606	0.546
1.3	16.33	2094	0.555
1.5	14.08	3245	0.566
2.0	11.21	1.6E5	0.566
2.5	8.20	6.8E5	0.646

Fig. 1. The Viterbi channel

Fig. 2. The concatenated channel

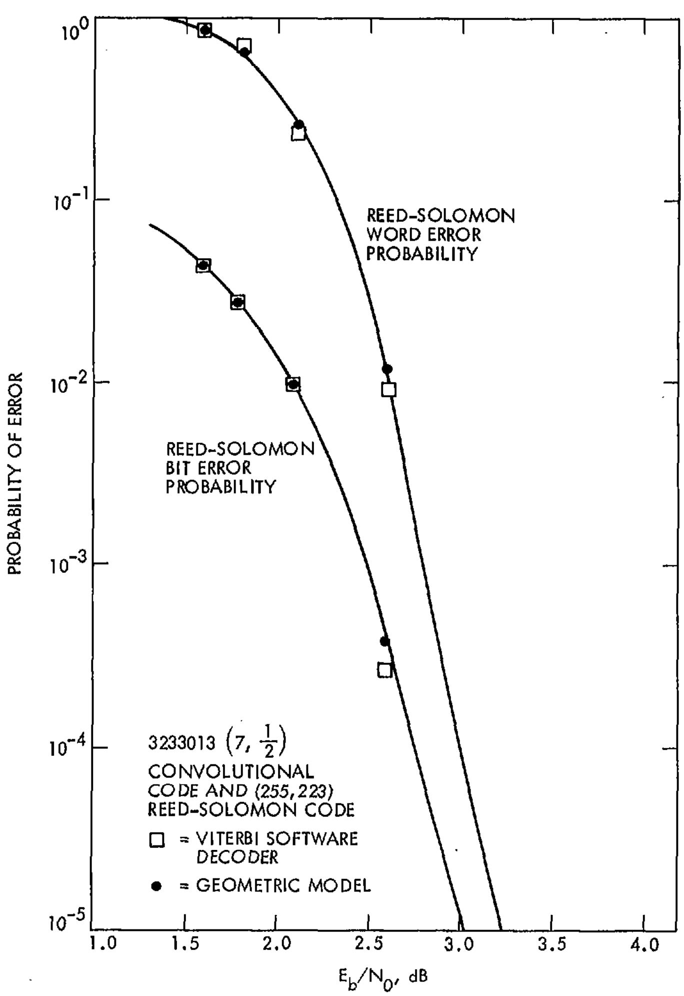

Fig. 3. Non-interleaved performance statistics for concatenated coding scheme assuming no system losses, (7, 1/2) convolutional code

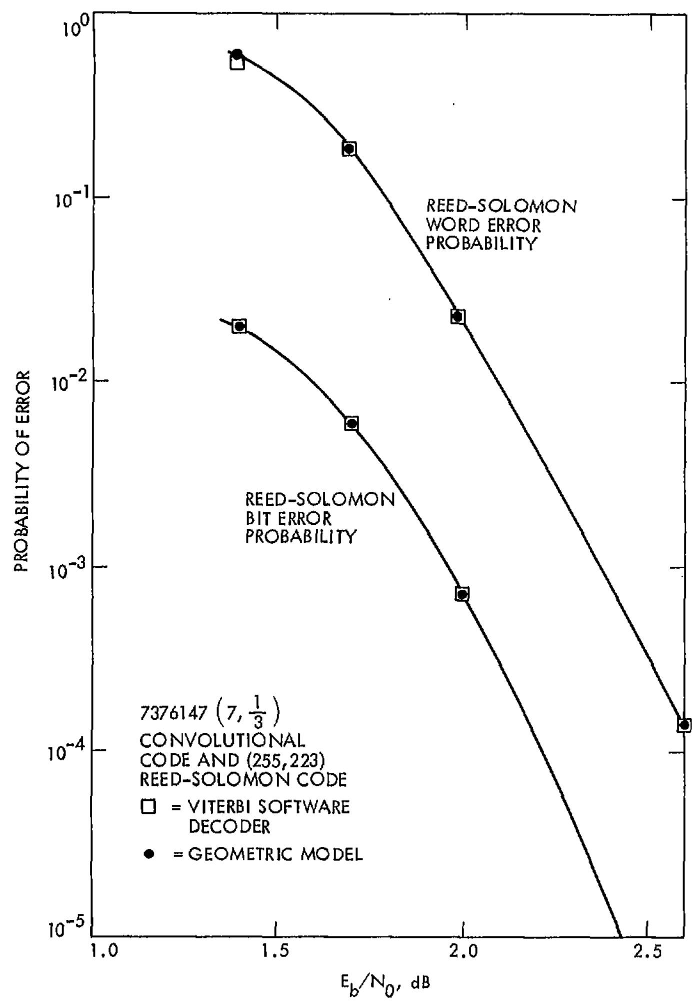

Fig. 4. Non-interleaved performance statistics for concatenated coding scheme assuming no system losses, (7, 1/3) convolutional code

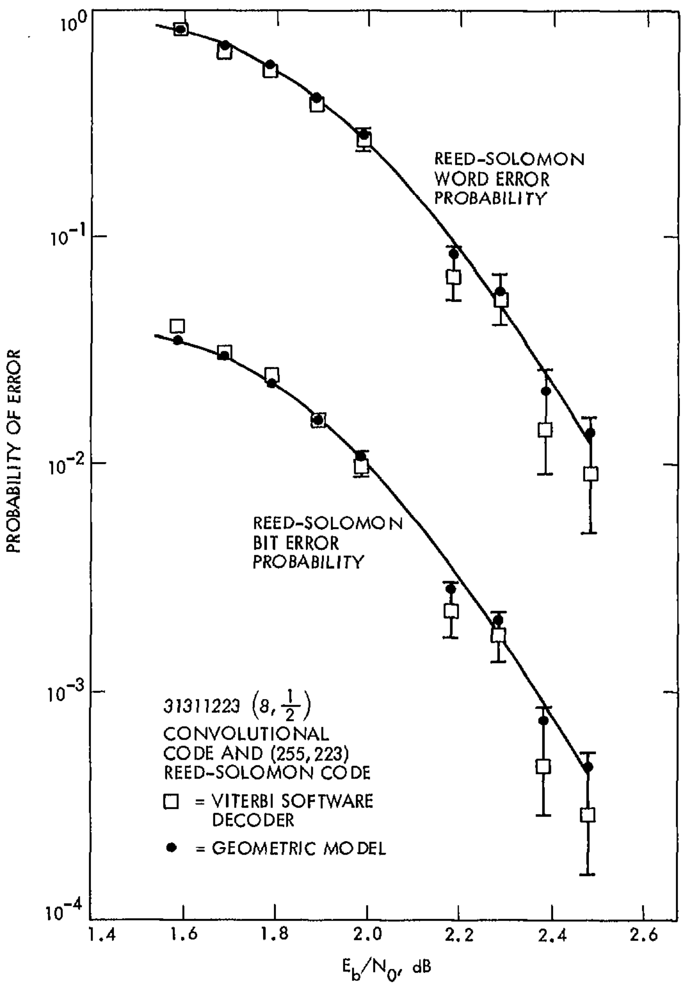

Fig. 5. Non-interleaved performance statistics for concatenated coding scheme assuming no system losses, (8, 1/2) convolutional code

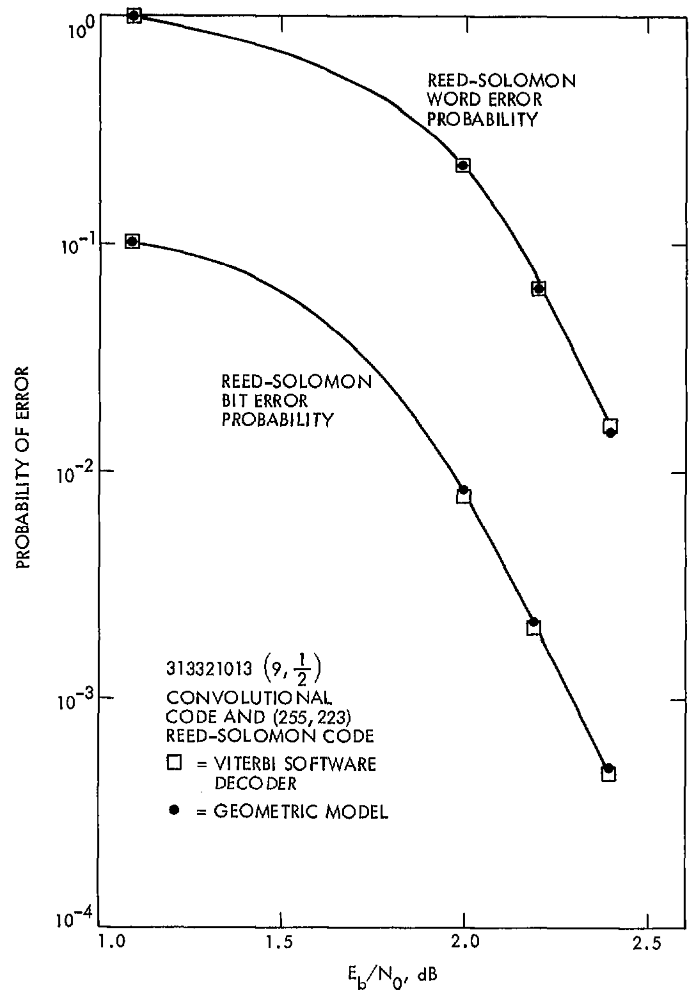

Fig. 6. Non-interleaved performance statistics for concatenated coding scheme assuming no system losses, (9, 1/2) convolutional code

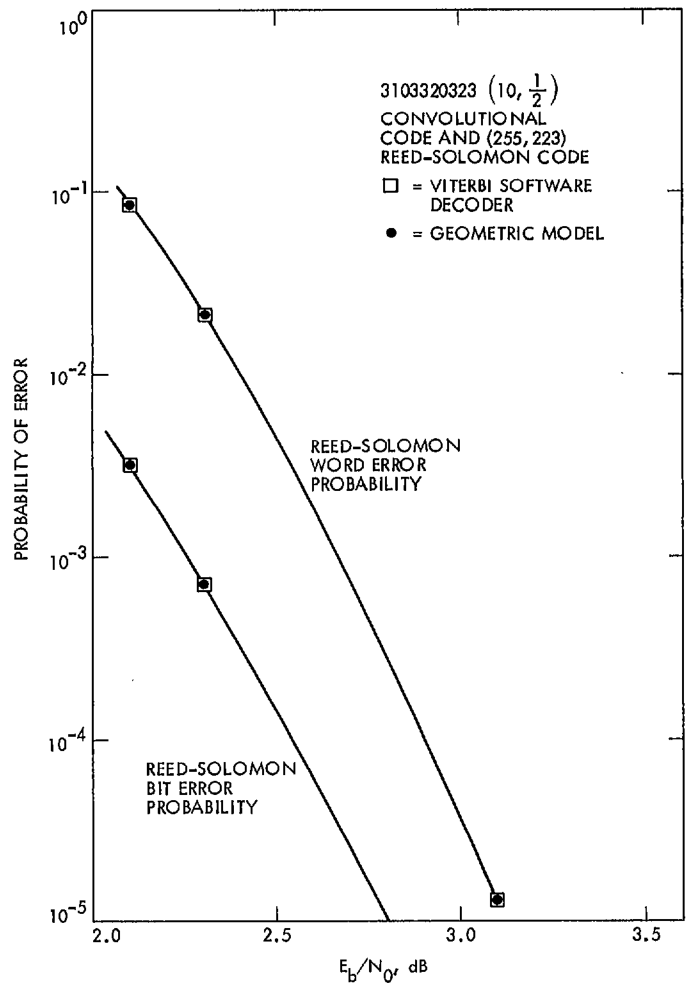

Fig. 7. Non-interleaved performance statistics for concatenated coding scheme assuming no system losses, (10, 1/2) convolutional code

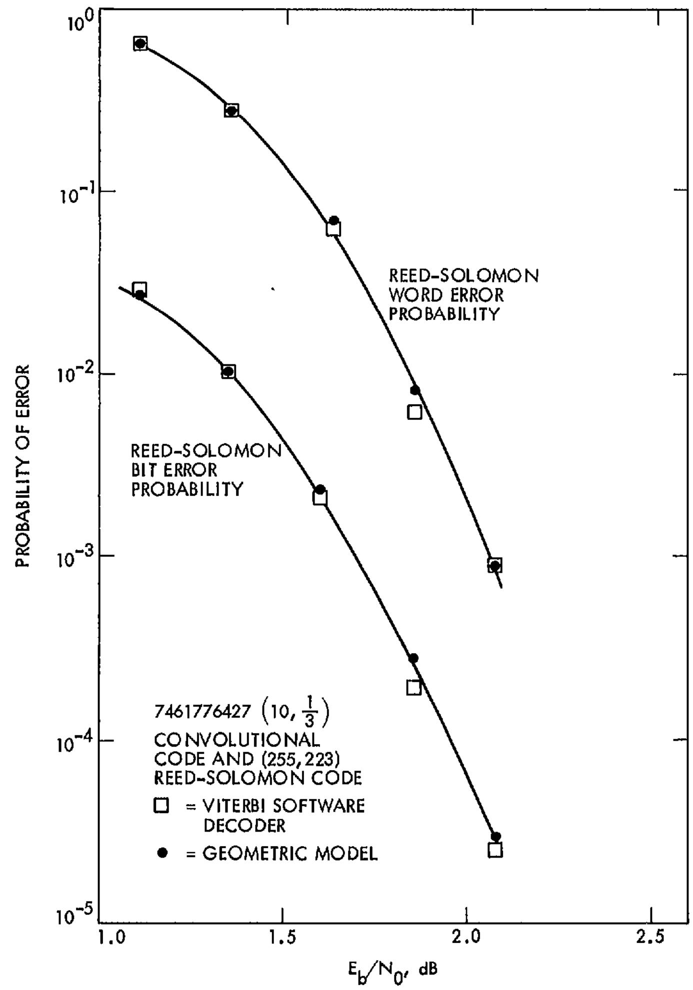

Fig. 8. Non-interleaved performance statistics for concatenated coding scheme assuming no system losses, (10, 1/3) convolutional code